UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/768,777	01/30/2004	Eric Justin Gould Bear	MSFT-3472 / 304032.02	7159

41505 7590 03/16/2007 WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION) CIRA CENTRE, 12TH FLOOR 2929 ARCH STREET PHILADELPHIA, PA 19104-2891	**EXAMINER** LONG, ANDREA NATAE

ART UNIT	PAPER NUMBER
2176	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/16/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

PTOL-90A (Rev. 10/06)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *30 January 2004*.

2a) ☐ This action is **FINAL**.   2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-41* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) *1-41* is/are rejected.

7) ☐ Claim(s) _____ is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☒ The drawing(s) filed on *30 January 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Claims 1-41 have been examined in response to application filed 01/30/2004 claiming

priority to Provisional application 60/433914, filed 12/16/2002.

Claim Rejections - 35 USC § 112

1.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly
> claiming the subject matter, which the applicant regards as his invention.

2.   Claims 2, 3, 37, 38, and 39 are rejected under 35 U.S.C. 112, second paragraph,

as being indefinite for failing to particularly point out and distinctly claim the subject

matter which applicant regards as the invention.

Claim 2 recites the limitation "the keyboard commands" in the first line of the

claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 3 recites the limitation "the Back key" in the first line of the claim. There is

insufficient antecedent basis for this limitation in the claim.

Claim 3 recites the limitation "the Escape key" in the first line of the claim. There

is insufficient antecedent basis for this limitation in the claim.

Claim 37 recites the limitation "the first such key" in the third paragraph of the

claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 37 recites the limitation "the last keyboard key" in the third and fourth

paragraph of the claim. There is insufficient antecedent basis for this limitation in the

claim.

Claim 38 recites the limitation "the keyboard commands" in the first line of the

claim. There is insufficient antecedent basis for this limitation in the claim.

Claim 39 recites the limitation "the Back key" in the first line of the claim. There

is insufficient antecedent basis for this limitation in the claim.

Claim 39 recites the limitation "the Escape key" in the second line of the claim.

There is insufficient antecedent basis for this limitation in the claim.

Claim Rejections - 35 USC § 101

3.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 37-39 are rejected under 35 U.S.C. 101 because the claimed invention is

directed to non-statutory subject matter.

As to independent claim 37 recites a "computer-readable medium having

computer-readable instructions. Applicant's specification states that program code can

be transmitted over some transmission medium, such as over electrical wiring or

cabling, through fiber optics (signals), or via any other form of transmission. Signals do

not fall within one of the four categories (process, machine, manufacture, or

composition of matter) of patent eligible subject matter.

Claims 38 and 38 are rejected for inheriting the deficiencies of independent claim

37.

Claim Rejections - 35 USC § 103

4.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

5.   **Claims 1-3, 35, and 37-39 are rejected under 35 U.S.C. 103(a) as being**

unpatentable over Hinckley et al (Pub. No. US 2001/0011995 A1), hereinafter

"Hinckley" in view of Inatomi (US Patent 5598522), hereinafter "Inatomi".

As to independent claim 1, Hinckley teaches a method for combining the

functionality a set of at least two commands calls into a single logical button (page 1

paragraph [0009] → taught as controls on an input device having more than 1 function

depending on the current application in which the commands are called). However

Hinckley does not teach wherein the commands are prioritized and recognizing when a

command is rejected. Inatomi teaches prioritizing a set of command calls from highest

to lowest priority (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59 →

taught as prioritizing the commands corresponding to frequency of use or the previous

command executed). Inatomi also teaches recognizing when a command call is

rejected by an application and sending at least one lower priority command call (column

5 lines 16-51 → taught as an application recognizing the users actions to reject the current command and to execute another command within the list).

It would have been obvious to one skilled in the art at the time the invention was made to have combined the combining of commands of Hinckley with the prioritizing of Inatomi to increase the amount of information that an input device can provide to the computer (page 1 paragraph [0009], Hinckley) and by simplifying the selecting of operations regarding the commands representing processes to be executed, the operability of the computer system can be improved (column 1 lines 22-25, Inatomi); further, providing a command processing system in which a process for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi).

As to dependent claim 2, Hinckley as modified by Inatomi teaches prioritizing multiple commands in a logical button. Hinckley teaches an input device such as a keyboard having input signals using transducers to implement functions. However, Hinckley as modified by Inatomi does not teach keyboard command therein combined are Back and Escape. It is well know to one skilled in the art that there is a relationship between the commands Back and Escape of a keyboard. Back is used to clear the previous textual input of a user, while Escape is well known to back out or close a determined screen/application. It would have been obvious to one skilled in the art at the time the invention was made to have combined the commands of Back and Escape

to reduce the real estate of keys on a keyboard and to enhance the functionally of commands within different applications.

As to dependent claim 3, Inatomi teaches prioritizing commands based on the previous executed command or on the frequency of use of a command (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59). Note the discussion in claim 2, about the combining of Back and Escape. While Hinckley or Inatomi don't explicitly teach the Back key being prioritized first and the Escape key being prioritized last, it is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi).

As to independent claim 35, Hinckley teaches a using an operating system shell hook to issue a command as an application command to an application (page 10 paragraph [0117]). However, Hinckley does not teach if the application command is rejected, issuing a different command directly to the operating system. Inatomi teaches recognizing when a command call is rejected by an application and sending at least one lower priority command call (column 5 lines 16-51 → taught as an application recognizing the users actions to reject the current command and to execute another command within the list).

It would have been obvious to one skilled in the art at the time the invention was

made to have combined message hook of Hinckley with the recognizing of an

application rejection of Inatomi to increase the amount of information that an input

device can provide to the computer (page 1 paragraph [0009], Hinckley) and by

simplifying the selecting of operations regarding the commands representing processes

to be executed, the operability of the computer system can be improved (column 1 lines

22-25, Inatomi), further, providing a command processing system in which a process for

a command can be executed with simple operations as much as possible under the GUI

circumstances (column 2 lines 55-58, Inatomi).

As to independent claim 37, Hinckley teaches combining the functionality a set

of at least two command calls into a single logical button (page 1 paragraph [0009] →

taught as controls on an input device having more than 1 function depending on the

current application in which the commands are called) and using an operating system

shell hook to make an application command call for the command calls (page 10

paragraph [0117]). However, Hinckley does not teach prioritizing commands. Inatomi

teaches executing in priority order until the first such key is not rejected, and therefore

accepted by the application, or until all but the last keyboard key remains and if all but

the last keyboard key are rejected by the application in the previous element, then

issuing the last keyboard key as a operating system command (column 4 lines 33-38,

59-67, column 5 lines 1-3, 17-32, 54-67, column 6 lines 1-8).

It would have been obvious to one skilled in the art at the time the invention was made to have combined the combining of commands of Hinckley with the prioritizing of Inatomi to increase the amount of information that an input device can provide to the computer (page 1 paragraph [0009], Hinckley) and by simplifying the selecting of operations regarding the commands representing processes to be executed, the operability of the computer system can be improved (column 1 lines 22-25, Inatomi), further, providing a command processing system in which a process for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi).

As to dependent claim 38, Hinckley as modified by Inatomi teaches prioritizing multiple commands in a logical button. Hinckley teaches an input device such as a keyboard having input signals using transducers to implement functions. However, Hinckley as modified by Inatomi does not teach keyboard command therein combined are Back and Escape. It is well know to one skilled in the art that there is a relationship between the commands Back and Escape of a keyboard. Back is used to clear the previous textual input of a user, while Escape is well known to back out or close a determined screen/application. It would have been obvious to one skilled in the art at the time the invention was made to have combined the commands of Back and Escape to reduce the real estate of keys on a keyboard and to enhance the functionally of commands within different applications.

As to dependent claim 39, Inatomi teaches prioritizing commands based on the previous executed command or on the frequency of use of a command (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59). Note the discussion in claim 38, about the combining of Back and Escape. While Hinckley or Inatomi don't explicitly teach the Back key being prioritized first and the Escape key being prioritized last, it is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi).

6.   **Claims 4-20, 23-24, and 26-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hinckley as modified by Inatomi in view of Snyder et al (Microsoft Windows 98 Keyboard Guide, 1999), hereinafter "Snyder".**

As to dependent claims 4, 5, 6, and 7, Hinckley as modified by Inatomi teaches prioritizing commands. Inatomi teaches prioritizing commands based on the previous executed command or on the frequency of use of a command (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process, for a command can be executed with

simple operations as much as possible under the GUI circumstances (column 2 lines

55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the

command APPCOMMAND_UP or keyboard commands Up Arrow, Scroll Up, or Page

Up. Snyder teaches common keyboard commands of up commands Up Arrow (pages

6, 8, 9, 10), Scroll Up (page 11 → taught as using the up arrow key to scroll a window),

and Page Up (page 11). While the APPCOMMAND_UP is not explicitly stated, it is well

known to one skilled in the art that Microsoft Windows is a command base software

which has code written in some type of programming language such as C++ or Java. It

is also well known that code is written in a simple and descriptive function of the

function at which it is intended to perform. Therefore it is suggestively obvious to one

skilled in the art to infer that the command APPCOMMAND_UP would be a suitable and

well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the

commands of Snyder with the prioritizing of commands of Hinckley as modified by

Inatomi to make typing easier for people who have difficulty using the keyboard or

pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 8, 9, 10, and 11, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process, for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the command APPCOMMAND_DOWN or keyboard commands Down Arrow, Scroll Down, or Page Down. Snyder teaches common keyboard commands of down commands Down Arrow (pages 6, 8, 9, 10), Scroll Down (page 11 → taught as using the down arrow key to scroll a window), and Page Down (page 11). While the APPCOMMAND_DOWN is not explicitly stated, it is well known to one skilled in the art that Microsoft Windows is a command base software which has code written in some type of programming language such as C++ or Java. It is also well known that code is written in a simple and descriptive function of the function at which it is intended to perform. Therefore it is suggestively obvious to one skilled in the art to infer that the command APPCOMMAND_DOWN would be a suitable and well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the commands of Snyder with the prioritizing of commands of Hinckley as modified by Inatomi to make typing easier for people who have difficulty using the keyboard or pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 12, 13, and 14, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one

skilled in the art to infer that any order of prioritizing can be achieved depending of the

frequency of use of a command on the previous executed command which will provide

a command processing system in which a process, for a command can be executed

with simple operations as much as possible under the GUI circumstances (column 2

lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the

command APPCOMMAND_NEXT or keyboard commands Tab or Right Arrow. Snyder

teaches common keyboard commands of next commands Tab (pages 3, 12, 17), and

Right Arrow (3, 5, 7, 8, 9, 10). While the APPCOMMAND_NEXT is not explicitly stated,

it is well known to one skilled in the art that Microsoft Windows is a command base

software which has code written in some type of programming language such as C++ or

Java. It is also well known that code is written in a simple and descriptive function of

the function at which it is intended to perform. Therefore it is suggestively obvious to

one skilled in the art to infer that the command APPCOMMAND_NEXT would be a

suitable and well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the commands of Snyder with the prioritizing of commands of Hinckley as modified by Inatomi to make typing easier for people who have difficulty using the keyboard or pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of operations regarding the commands representing processes to be executed, thereby the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 15, 16, and 17, Hinckley as modified by Inatomi teaches prioritizing commands. Inatomi teaches prioritizing commands based on the previous executed command or on the frequency of use of a command (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process, for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the command APPCOMMAND_PREV or keyboard commands Shift-Tab or Left Arrow. Snyder teaches common keyboard commands of previous commands Shift-Tab (page 12) and Left Arrow (pages 3, 5, 7, 8, 9, 10). While the APPCOMMAND_PREV is not explicitly stated, it is well known to one skilled in the art that Microsoft Windows is a command base software which has code written in some type of programming language such as C++ or Java. It is also well known that code is written in a simple and

descriptive function of the function at which it is intended to perform. Therefore it is

suggestively obvious to one skilled in the art to infer that the command

APPCOMMAND_PREV would be a suitable and well-known command naming in the

keyboard commands.

It would have been obvious to one skilled in the art to have combined the

commands of Snyder with the prioritizing of commands of Hinckley as modified by

Inatomi to make typing easier for people who have difficulty using the keyboard or

pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 18, 19, and 20, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one

skilled in the art to infer that any order of prioritizing can be achieved depending of the

frequency of use of a command on the previous executed command which will provide

a command processing system in which a process, for a command can be executed

with simple operations as much as possible under the GUI circumstances (column 2

lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the

command APPCOMMAND_ENTER or keyboard commands Return or Enter. Snyder

teaches common keyboard commands of enter commands Return (page 3, 11, 16, 17,

18) and Enter (pages 6, 7, 12). While the APPCOMMAND_ENTER is not explicitly

stated, it is well known to one skilled in the art that Microsoft Windows is a command

base software which has code written in some type of programming language such as

C++ or Java. It is also well known that code is written in a simple and descriptive

function of the function at which it is intended to perform. Therefore it is suggestively

obvious to one skilled in the art to infer that the command APPCOMMAND_ENTER

would be a suitable and well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the

commands of Snyder with the prioritizing of commands of Hinckley as modified by

Inatomi to make typing easier for people who have difficulty using the keyboard or

pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 22, 23, 24, and 26, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one

skilled in the art to infer that any order of prioritizing can be achieved depending of the

frequency of use of a command on the previous executed command which will provide

a command processing system in which a process, for a command can be executed

with simple operations as much as possible under the GUI circumstances (column 2

lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the

command APPCOMMAND_OUT or keyboard commands Browser Back, Escape or Alt-

F4. Snyder teaches common keyboard commands of out commands Browser Back

(page 12), Escape (pages 6,12) and Alt-F4 (page 4). While the APPCOMMAND_OUT

is not explicitly stated, it is well known to one skilled in the art that Microsoft Windows is

a command base software which has code written in some type of programming

language such as C++ or Java. It is also well known that code is written in a simple and

descriptive function of the function at which it is intended to perform. Therefore it is

suggestively obvious to one skilled in the art to infer that the command

APPCOMMAND_OUT would be a suitable and well-known command naming in the

keyboard commands.

It would have been obvious to one skilled in the art to have combined the

commands of Snyder with the prioritizing of commands of Hinckley as modified by

Inatomi to make typing easier for people who have difficulty using the keyboard or

pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 27, 28, 29, 30, and 31, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one

skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process, for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the command APPCOMMAND_SWITCH or keyboard commands Alt-Escape, Alt-Tab, Windows Key, or Home. Snyder teaches common keyboard commands of switch commands Alt-Escape (page 4), Alt-Tab (page 4), Windows Key (page 1), and Home (page 7). While the APPCOMMAND_SWITCH is not explicitly stated, it is well known to one skilled in the art that Microsoft Windows is a command base software which has code written in some type of programming language such as C++ or Java. It is also well known that code is written in a simple and descriptive function of the function at which it is intended to perform. Therefore it is suggestively obvious to one skilled in the art to infer that the command APPCOMMAND_SWITCH would be a suitable and well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the commands of Snyder with the prioritizing of commands of Hinckley as modified by Inatomi to make typing easier for people who have difficulty using the keyboard or pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of operations regarding the commands representing processes to be executed, thereby the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

As to dependent claims 32, 33, and 34, Hinckley as modified by Inatomi

teaches prioritizing commands. Inatomi teaches prioritizing commands based on the

previous executed command or on the frequency of use of a command (column 4 lines

33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one

skilled in the art to infer that any order of prioritizing can be achieved depending of the

frequency of use of a command on the previous executed command which will provide

a command processing system in which a process, for a command can be executed

with simple operations as much as possible under the GUI circumstances (column 2

lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the

command APPCOMMAND_MENU or keyboard commands Shift-F10 or special

command for Settings. Snyder teaches common keyboard commands of menu

commands Shift-F10 (page 6) and special commands for Settings (pages 10, 13).

While the APPCOMMAND_MENU is not explicitly stated, it is well known to one skilled

in the art that Microsoft Windows is a command base software which has code written in

some type of programming language such as C++ or Java. It is also well known that

code is written in a simple and descriptive function of the function at which it is intended

to perform. Therefore it is suggestively obvious to one skilled in the art to infer that the

command APPCOMMAND_MENU would be a suitable and well-known command

naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the

commands of Snyder with the prioritizing of commands of Hinckley as modified by

Inatomi to make typing easier for people who have difficulty using the keyboard or

pressing more that more key at a time (page 1, Snyder) and to simplify the selecting of

operations regarding the commands representing processes to be executed, thereby

the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

7.   **Claims 36, 40, and 41 are rejected under 35 U.S.C. 103(a) as being**

unpatentable over Hinckley as modified by Inatomi, in further view of Logitech

(LogitechMouse User's Guide, 1993), hereinafter "Logitech".

        As to independent claim 36, 40, and 41, Hinckley as modified by Inatomi

teaches the method of claim 1, note the discussion above. However Hinckley as

modified by Inatomi does not teach ENTER, UP, DOWN, and OUT commands being

applied to an object and a subsystem for processing the commands. Logitech teaches

a system comprising an interface (MouseWare software and MouseMan) that generates

a logical input for one of a group of commands to be applied to an object (page 11 →

taught as using the MouseWare to assign shortcuts to the MouseMan and using the

MouseMan to select and manipulate objects). Logitech further teaches a subsystem for

processing an ENTER, UP (page up), DOWN (page down), and OUT (close application)

command (page 11 → taught as using the software with a computer). It is well known

that computers use a processor to execute computer instructions. Logitech teaches

using the software in conjunction with a computer for functionality. It is reasonable to

one skilled in the art that the subsystem (computer) would be used to process the

ENTER, UP, DOWN, and OUT command.

Therefore it would have been obvious to one skilled in the art at the time the invention was made to have combined the method of Hinckley as modified by Inatomi with the commands of Logitech to simplify the selecting of operations regarding the commands representing processes to be executed, thereby the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

8.   **Claims 21 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hinckley as modified by Inatomi in further view of Stern et al (Quicktime 5 for Macintosh and Windows: Visual Quickstart Guide, 2001), hereinafter "Stern".**

As to dependent claims 21 and 25, Hinckley as modified by Inatomi teaches prioritizing commands. Inatomi teaches prioritizing commands based on the previous executed command or on the frequency of use of a command (column 4 lines 33-38, line 66 through column 5 line 3, lines 55-59). It is suggestively obvious for one skilled in the art to infer that any order of prioritizing can be achieved depending of the frequency of use of a command on the previous executed command which will provide a command processing system in which a process, for a command can be executed with simple operations as much as possible under the GUI circumstances (column 2 lines 55-58, Inatomi). However, Hinckley as modified by Inatomi does not teach the commands APPCOMMAND_ENTER, APPCOMMAND_OUT or special commands Play and Stop. Stern teaches mouse and keyboard commands operable to QuickTime Player to Play and Stop. While the APPCOMMAND ENTER and APPCOMMAND_OUT

is not explicitly stated, it is well known to one skilled in the art that QuickTime Player is command base music software which has code written in some type of programming language such as C++ or Java. It is also well known that code is written in a simple and descriptive function of the function at which it is intended to perform. Therefore it is suggestively obvious to one skilled in the art to infer that the commands APPCOMMAND_ENTER and APPCOMMAND_OUT would be a suitable and well-known command naming in the keyboard commands.

It would have been obvious to one skilled in the art to have combined the commands of Stern with the prioritizing of commands of Hinckley as modified by Inatomi to allow users to perform useful functions within multimedia programs with a standard controller and to simplify the selecting of operations regarding the commands representing processes to be executed, thereby the operability of the computer system can be improved (column 1 lines 22-25, Inatomi).

Conclusion

9.   The prior art made of record on Form PTO 892 and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andrea N. Long whose telephone number is 571-270-1055. The examiner can normally be reached on Mon - Thurs 6:00 am to 3:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather Herndon can be reached on 571-272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Andrea Long
03/07/2007

WILLIAM BASHORE
PRIMARY EXAMINER